IN THE SPECIFICATION

5   1. On page 1, line 1, please amend the Title of the Application as follows:

-- PROGRAM CONTROLLED EMBEDDED-DRAM-DSP ARCHITECTURE AND

METHODS --

10   2. On page 56, line 1, please amend the Title of the Application as follows:

-- PROGRAM CONTROLLED EMBEDDED-DRAM-DSP ARCHITECTURE AND

METHODS --

15   3.   On page 56, lines 3 through 20, please delete the existing text in its entirety, and

replace it with the following text:

> -- An efficient embedded-DRAM processor architecture and associated methods.
> In one exemplary embodiment, the architecture includes a DRAM array, a set of
> register files, set of functional units, and a data assembly unit. The data assembly
> unit includes a set of row-address registers and is responsive to commands to
> activate and deactivate DRAM rows and to control the movement of data
> throughout the system. A pipelined data assembly approach allowing the
> functional units to perform register-to-register operations, and allowing the data
> assembly unit to perform all load/store operations using wide data busses. Data
> masking and switching hardware allows individual data words or groups of words
> to be transferred between the registers and memory. Other aspects of the invention
> include a memory and logic structure and an associated method to extract data
> blocks from memory to accelerate, for example, operations related to image
> compression and decompression.--

4.   On page 23, lines 1-30 of the specification as filed, please amend the text as

follows:

> --A set of three high-speed register files 112, 114, and 116 are connected to the mask
> and switch unit 108, also preferably via dw1-word wide data busses. In alternate
> embodiments, rows of width dw1 may be sub-divided and sent to smaller register files,
> or can be multiplexed and sent to the register files in a plurality of transfer cycles. The
> register files 112, 114, and 116 are preferably implemented using high speed SRAM
> technology and are each coupled to a selector 120 which in turn couples the register

files 112, 114, 116 to the set of functional units 128. While the preferred embodiment employs three high-speed register files 112, 114, 116, systems with other numbers of register files are anticipated. To implement aspects of the present invention, at least two high-speed register files 112, 114 should be used. A data assembly unit 122 is coupled via address and control lines 118 to the high-speed register files 112, 114, and 116. In some embodiments, additional data paths may be used to transfer data between internal registers located within the data assembly unit 122 and registers located within the register files 112, 114 and 116. The data assembly unit 122 is also coupled via control and address lines 123 to the mask and switch unit 108. Address information delivered to the mask and switch unit 108 from the data assembly unit 122 is further coupled to the address and control inputs of the DRAM array modules 102, 104, 106 as well as to the DMA/SAM 110. The set of functional units 128 optionally receive program instructions as selected by a multiplexer 132. The multiplexer 132 has one input coupled to an interleaved DRAM program memory array [[134]] via a set of lines [[124]] 126 and the mask and switch unit 108. The multiplexer 132 has another input coupled to an output of a branch-oriented instruction cache 124. The program memory DRAM array [[134]] is preferably implemented with a dw3 width data bus, where dw3 represents the number of instructions to be prefetched into the a prefetch buffer (not shown). The prefetch buffer holds instructions to be executed by the functional units 128. In some implementations, the prefetch buffer may also contain instructions to be executed by the data assembly unit 122 as well. The program memory array 134 is also preferably stacked into an interleaved access bank so that one fetch packet containing instructions may be fetched per clock cycle when instructions are fetched from a sequential set of addresses. As will be discussed below in connection with FIG. 5, the program DRAM 134 may also preferably --

5. On page 28, lines 1-17 of the specification as filed, please amend the text as follows:

-- FIG. 2 shows one embodiment of the invention highlighting the data transfer and register selection mechanisms 200 between the DRAM arrays 102 and, for example, the register file 112. The connections to the other register files 114, 116 are similar. The register file 112 and is coupled to a set of switches 204. Each of the switches 204 includes a first port coupling to the register file 112, a second port coupling to a parallel load/store channel carrying a masked DRAM row 208 to or from the mask and switch unit 108 via an interface 214. Each switch 204 also includes a second port coupling to a selector switch 206. The selector switch 206 selectively couples the registers of the register file 112 either to the functional units 128 or to the data assembly unit 122. Specifically, the second port of the selector switch 206 couples the registers 112 to an optional inter-register move unit 224 included within the data assembly unit 122. The data assembly unit 122 also includes a load/store unit 226. The load/store unit 226 presents a mask switch control input 230 to the mask and switch unit 108. The load/store unit 226 also presents a row-address input 228 to the mask and switch unit 108. In some embodiments, the row address control 228 may pass directly to the DRAM arrays 102, 104, 106. In the embodiment shown, the mask and switch unit 108 performs address decoding functions as well as its other tasks.--